(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: 0 376 271 A2

(12) EUROPEAN PATENT APPLICATION

(21) Application number: 89123987.3

(51) Int. Cl.5: G06K 15/02

(22) Date of filing: 27.12.89

(30) Priority: 26.12.88 JP 326265/88

(43) Date of publication of application:
04.07.90 Bulletin 90/27

(84) Designated Contracting States:
DE FR GB

(71) Applicant: Oki Electric Industry Company, Limited
7-12, Toranomon 1-chome Minato-ku
Tokyo 105(JP)

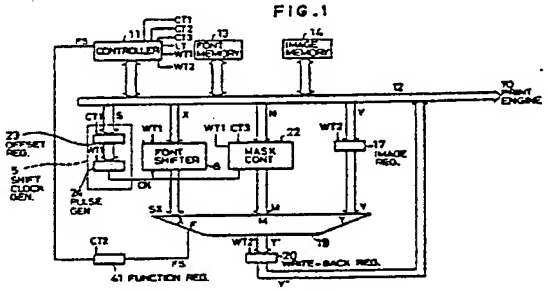
(72) Inventor: Suzuki, Masahiro c/o Oki Electric Industry Co. Ltd
7-12, Toranomon 1-chome
Minato-ku Tokyo(JP)
Inventor: Nagata, Masato c/o Oki Electric Industry Co. Ltd
7-12, Toranomon 1-chome
Minato-ku Tokyo(JP)

(74) Representative: Betten & Resch
Reichenbachstrasse 19
D-8000 München 5(DE)

(54) Print data generator.

(57) A print data generator uses bit-mapped font data to modify bit-mapped image data by shifting a series of font words followed by an all-zero word through a bit offset S and performing an operation on those bits of image words and shifted font words that correspond to enable bits in converted mask control words. To create the converted mask control words, the print data generator has a mask control data source for generating source mask control words consisting of S disable bits followed exclusively by enable bits, and a mask control data converter that passes the first source mask control word through unaltered, substitutes all enable bits for the next (N - 2) source mask control words, and inverts the Nth source mask control word. By generating and converting mask control words automatically, the print data generator creates print data rapidly without overloading its controller.

FIG.1



EP 0 376 271 A2

a font shifter (6) for receiving (N - 1) successive font words from said font memory and then an all-zero word, receiving said series of S shift clock pulses from said shift clock generator, and generating shifted font words.
the first shifted font word in each line consisting of S bits at the tail of the first font word for the S bits at the head of the shifted font word and zero bits for the remaining bits;
the last shifted font word in each line consisting of (L - S) bits at the head of the last font word for the (L -S) bits at the tail of the shifted font word and zero bits for the remaining bits;
the shifted font words other than the first and last shifted font words each consisting of (L - S) bits at the head of one received font word for the tail of the shifted font word and S bits at the tail of the preceding received font word for the head of the shifted font word;
a mask control data source (7) for receiving said series of S shift clock pulses from said shift clock generator and generating source mask control words comprising S disable bits followed exclusively by enable bits;
a mask control data converter (8) for receiving data indicative of the width number N from said controller, receiving said source mask control words from said mask control data source, and generating converted mask control words by passing the first source mask control word through unaltered, substituting all enable bits for the next (N - 2) source mask control words, and inverting the Nth source mask control word; and
an arithmetic-logic unit (19) for receiving an image word from said image memory, a shifted font word from said font shifter, and a converted mask control word from said mask control data converter, and using said font word to modify bits of said image word that correspond to enable bits in said converted mask control word, thereby creating a modified image word; and
means (20, 12) for transferring said modified image word to said image memory.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a block diagram of a print data generator.

Fig. 2 is a more detailed block diagram of the font shifter and the mask controller in Fig. 1.

Fig. 3A shows an example of print-out of part of one page.

Fig. 3B shows how the image words are arranged in the image memory.

Fig. 4A shows an example of font data in a dot matrix representation.

Fig. 4B shows how the font words are arranged in the font memory.

Fig. 5 shows how a font data is superimposed on an image data.

Fig. 6A to Fig. 6D show the examples of the contents in the image memory and the font memory, and how the respective words are combined.

Fig. 7 shows selective enablement and disablement of the operation at the arithmetic-logic unit.

Fig. 8 is a time chart showing signals produced by the controller.

Fig. 9 is a time chart showing word timing signals and shift clocks.

DETAILED DESCRIPTION OF THE INVENTION

A novel print data generator for modifying bit-mapped image data according to bit-mapped font data will be described with reference to the drawings. The image data and modifying data, such as font data, both comprise words consisting of a certain number, L, of bits, such as eight, sixteen, or thirty-two bits. In the drawings, words will be shown as consisting of eight bits. In the following description, the modifying data are assumed to be font data, but the invention is applicable where the modifying data are other than font data.

Overall Operation

Referring now to Fig. 1 and Fig. 2, words of image data are stored in an image memory 14, and words of font data are stored in a font memory 13. The font data is read from the font memory 13 and the image data is read from the image memory 14, operation is performed on the image data and the font data by use of an arithmetic operation unit 19, and write the result of the operation, i.e., the modified image data into the image memory 14. The modified image data are supplied to a print engine for printing.

More specifically, the image memory 14 and the font memory 13 are connected to a bus 12 for the exchange of data a word at a time with other parts of the print data generator. The image data read from the image memory 14 are stored in an image register 17, while the font data read from the font memory 13 is supplied to a font shifter 6, where it is shifted by a predetermined number of bit positions for alignment with the image data. The operation on the image data Y and the font data X for modifying the image data Y is performed by an arithmetic-logic unit (ALU) 19, and the result of the operation, the modified image data Y′, are stored in a write-back register 20. It should be noted that the modified image data Y′ obtained at the output of the the arithmetic-logic unit 19 is written in the

PRINT DATA GENERATOR

BACKGROUND OF THE INVENTION

This invention relates to a print data generator for generating bit-mapped image data in a device such as a page printer.

Such a device generally has an image memory for storing bit-mapped image data, and a modifying data memory, typically a font memory, formed for example of a RAM or a ROM, for storing the bit maps of characters and symbols. To print a character or symbol, its bit map is read from the font memory a word at a time, similar words of image data are read from the image memory, and an operation is performed on the font words and corresponding image words to modify the image words. The modified image words are then written back to the same memory location in the image memory from which the particular image words have been read. The modified image words are then supplied to a print engine, not shown, for printing. The image data may already contain font data developed therein, or data representing graphics, rule marks, background pictures, etc. The operation for the modification of the image data may be a logical sum (OR) operation, a logical product (AND) operation, or an exclusive logical sum (ex-OR) operation on the data input at the X input terminal and the Y input terminal, or selection of one of the data at tile X input terminal or the data at the Y input terminal. When a "transparent printing," in which neither of the black dots in the image data and the black dots in the font data are suppressed, is to be performed, the logical sum operation is performed. When a "non-transparent printing," in which the image data is suppressed in the area where the font data is superimposed, and the bits of the font data become the bits of the image data, is to be performed, the the selection of the bits of the font data is performed.

The process is complicated by the fact that the font words may be offset from the image words by a certain number of bits. Before the operation is performed, the font words must then be shifted to align them with the image words. A by-product of this shift is that in each line of image data, the initial bits of the first image word and the final bits of the last image word are not covered by font data. The operation must therefore be disabled on the uncovered bits to avoid modifying image data outside the width of the character or symbol. Control words comprising enable bits and disable bits are used for this purpose.

The data generation process is overseen by a controller such as a microprocessor which specifies the addresses of the image data and font data in their respective memories, and the bit offset between the font words and image words. The microprocessor may also shift the font data, perform the necessary modification of image words, and write the modified image words back to image memory. These tasks are time-consuming, however, and the controller has other tasks to perform as well, such as generating bit maps from mathematical descriptions and communicating with a host computer.

Specialized hardware has therefore been developed for generating print data. Such specialized hardware typically comprises a shift register for shifting the font words and a programmable arithmetic-logic unit for carrying out a specified operation on the font words and image words. By performing these tasks independently, such specialized hardware can significantly speed up the generation of print data while leaving the controller free to perform other tasks.

However, it has still been necessary for the controller to control the specialized hardware fairly closely. A particular problem is that the controller has to identify the first, intermediate, and last image words in each line and set corresponding mask control data in a register in the specialized hardware. Control of the specialized hardware thus still imposes a significant load on the controller and slows down the printing process, particularly on pages with large numbers of characters or symbols.

SUMMARY OF THE INVENTION

An object of the present invention is accordingly to generate print data without imposing a heavy load on the controller.

A particular object of this invention is to generate mask control data automatically.

A print data generator according to the invention is for modifying bit-mapped image data according to bit-mapped font data. It comprises:

an image memory (14) for storing image words, each word consisting of L bits;

a font memory (13) for storing font words each word consisting of L bits;

a controller (11) for selecting a bit offset number S representing a bit offset between said image words and said font words, and a width number W which is one greater than the number of font words per line;

a shift clock generator (5) for receiving said offset number S from said controller and repeatedly generating series of shift clock pulses comprising S pulses each;

position of each of the shifted font words SX differs depending on whether it is the first shifted word in each line, the last shifted word in each line, or other than the first and the last.

(a) The first shifted font word in each line consists of S bits at the tail of the first font word for the S bits at the head of the shifted font word and zero bits for the remaining bits.

(b) The last shifted font word in each line consists of (L - S) bits at the head of the last font word for the (L -S) bits (L being the number of bits per word) at the tail of the shifted font word and zero bits for the remaining bits.

(c) The shifted font words other than the first and last shifted font words each consists of (L - S) bits at the head of one received font word for the tail of the shifted font word and S bits at the tail of the preceding received font word for the head of the shifted font word.

The font shifter 6 comprises a font register 15 for receiving a font word X from the bus 12. In Fig. 2, it is shown to be storing a word "10110101."

The font word stored in the font register 15 is promptly transferred to a first shift register 25A.

A shift register is a device for storing a word of data and shifting it in a certain direction (to the right in the drawings). The word can be input to and output from the shift register in parallel, all bits at once, but the shift register also has a serial input terminal at one end (the left end in the drawings) and a serial output terminal at the other end (the right end in the drawings) for input and output of one bit at a time. The shift register furthermore has a clock terminal for receiving shift clock pulses. Each time a shift clock pulse is received, each bit of the stored word is shifted one position to, for example, the right, causing the rightmost bit to be shifted out through the serial output terminal and the leftmost bit position to be filled with a bit input through the serial input terminal.

The first shift register 25A receives a series of S shift clock pulses from the shift clock generator 5 and thereby shifts the stored font word to the right by S bit positions. Zero bits are stored in the vacated bit positions, by grounding the serial input terminal of the first shift register 25A, for example. The bits shifted out of the first shift register 25A are supplied to the serial input terminal of a second shift register 25B.

By means of the word timing signal WT1, the second shift register 25B is reset to all-zero data each time the first shift register 25A receives a font word. The second shift register 25B also receives the series of S = 3 shift clock pulses from the shift clock generator 5, which cause it to shift the all-zero data to the right by S bit positions. The vacated bit positions are filled with bits shifted out of the first shift register 25A. Fig. 2 shows a state in which the data transferred from the register 15 has been shifted by three bits. The shift register 25B has been reset to all-zero before the shift.

A tail register 26 is connected to receive the parallel output of the second shift register 25B. When the modified image word is transferred from the write-back register 20 to the image memory 14, the contents of the second shift register 25B are transferred in parallel to the tail register 26. This transfer takes place responsive to the word timing signal WT1.

The contents of the first shift register 25A and the tail register 26 are supplied to an OR-logic circuit 27, which performs an OR operation on them and sends the result, as the output of the font shifter 6, to the X input terminal of the arithmetic-logic unit 19. The OR-logic circuit 27 may comprise, for example, wired-OR connections of corresponding parallel bit output lines from the first shift register 25A and the tail register 26.

Image register 17

The image register 17 temporarily stores the image word Y read from the image memory 14 responsive to the timing signal WT2, and sends it to the Y input terminal of the arithmetic-logic unit 19. In Fig. 2, the data shown to be stored consists of a word "01010101."

Arithmetic-logic Unit 19

The arithmetic- logic unit 19 performs various operations. For instance, it performs a logical sum (OR) operation, a logical product (AND) operation, or an exclusive logical sum (ex-OR) operation on the data input at the X input terminal and the Y input terminal, or it selects one of the data at the X input terminal or the data at the Y input terminal, and outputs the selected data. When the "transparent printing," in which neither of the black dots in the image data and the black dots in the font data are suppressed, is to be performed, the logical sum operation is performed. When the "non-transparent printing," in which the image data is suppressed in the area where the font data is superimposed, and the bits of the font data become the bits of the image data, is to be performed, the data at the X input terminal is selected and output.

The arithmetic-logic unit 19 has a function specifying input terminal F through which function specifying data FS is input from a function register 41. The function specifying data FS is set in the register 41 by the controller 11 when the character timing signal CT2 is produced. The function speci-

fying data FS specifies an operation to be performed by the arithmetic-logic unit 19.

The arithmetic-logic unit 19 also has an input terminal M for receiving converted mask control word CM from a mask controller 22, to be described later. The mask control data designates whether the operation at the arithmetic-logic unit 19 should be performed on the respective bits. The net effect is that the arithmetic-logic unit 19 performs the operation specified by the function specifying data FS on those bits of the image word Y that correspond to enable bits in the converted mask control word CM.

Write-back Register 20

The write-back register 20 temporarily stores the output of the arithmetic-logic unit 19, that is a modified image word Y', when the word timing signal WT2' is produced. In Fig. 2, the modified image word Y' stored in the write-back register 20 is a word "0101011." This assumes that a logical sum operation has been performed by the arithmetic-logic unit 19 on the data input at the X and Y input terminals. The modified image word Y' is then transferred to the image memory 14 via the bus 12.

Mask Controller 22

The mask controller 22 which is provided to generate mask control words for enabling or disabling the operation on each bit in each image word. The mask controller 22 comprises a mask control data source 7 and a mask control data converter 8.

Mask Control Data Source 7

The mask control data source 7 receives the above mentioned series of S shift clock pulses from the shift clock generator 5 and generates source mask control words comprising S disable bits followed exclusively by enable bits.

The mask control data source 7 comprises a mask shift register 29, in which a word of all "1" is set each time the word timing signal WT1 is input, and a bit "0" is serially input each time a shift clock CK is input, and the data stored is shifted by one bit position, toward the right, as seen in Fig. 2. When S = 3 shift clock pulses have been received, the mask shift register 29 provides a source mask control word comprising S disable bits followed exclusively by enable bits. In the succeeding discussion, a disable bit will be a "0" bit and an enable bit will be a "1" bit. Thus, the mask shift register 29 serves to generate source mask control words that are used for selectively enabling and disabling the operation at the arithmetic-logic unit 19.

Mask Control Data Converter 8

The mask control data converter 8 receives data indicative of the width number N from the controller 11 and stores the width number N when the character timing signal CT3 is produced, and the mask control words from the mask control data source 7, and generates converted mask control words by passing the first mask control word through unaltered, substituting all enable bits for the next N - 2 mask control words, and inverting the Nth mask control word.

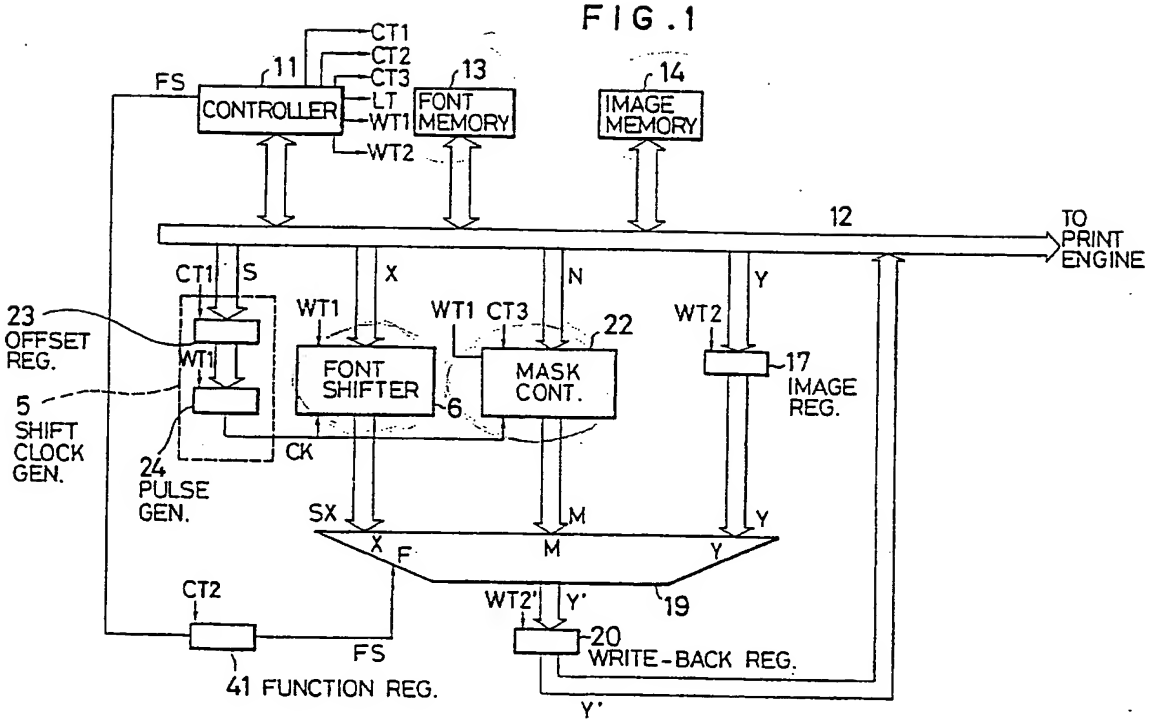
Width Register 21

The mask control word converter 8 comprises a width register 21 which receive the width number N from the controller 11, and stores the width number N when the character timing signal CT3 is produced. This number is one greater than the number of font words in a font, and signifies the number of the image words in each line partly or entirely superimposed with a font. For instance, where the character or font is as shown in Fig. 4A, the number of the words in each line is 2, and the number N stored in the width register 21 is 3. This means for performing the image data modifying operation, three words, rather than two words, must be read from the image memory 14, since the font word may be offset with respect to the image word, as shown in Fig. 5. In Fig. 5, the font words are offset by three bit positions and the font or the character area L is superimposed partially or entirely with three image words in each line.

Down Counter 28

The mask control word converter 8 also comprises a down counter 28. Responsive to the line timing signal LT produced at the beginning of processing of each line, the number N is transferred from the width register 21 to the down counter 28. Each time a word timing signal WT1 is input, the down counter 28 is decremented by one. When the word timing signal WT1 is input three times, the count value of the down counter 28 is reduced to "0." At the beginning of processing of the next line in the font data, the line timing signal LT is again supplied, and responsive to this line

FIG.1



EP 0 378 271 A2

FIG.4A

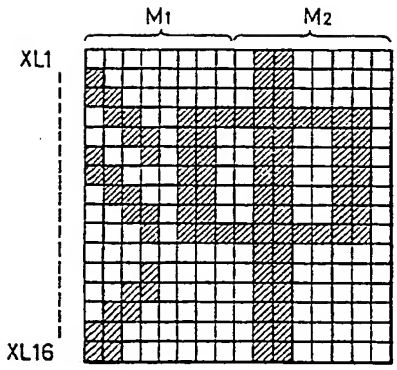
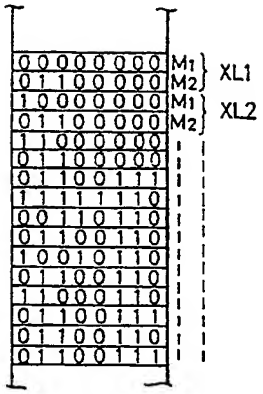


FIG.4B



EP 0 378 271 A2

032034-03

中华人民共和国国家知识产权局

邮政编码：100032 北京市西城区金融街19号富凯大厦B座11层 中原信达知识产权代理有限责任公司 谢丽娜	发文日期 专利局 2004.9.3.1 发文
申请号：031438296	
申请人：三星电子株式会社	
发明创造名称：用于在无线电话中显示字体的系统和方法	绝限 2005-01-18

第一次审查意见通知书

1. ☑应申请人提出的实审请求，根据专利法第35条第1款的规定，国家知识产权局对上述发明专利申请进行实质审查。
   ☐根据专利法第35条第2款的规定，国家知识产权局决定自行对上述发明专利申请进行审查。
2. ☑申请人要求以其在：
   KR 专利局的申请日 2002年08月06日为优先权日，
   专利局的申请日 年 月 日为优先权日，
   专利局的申请日 年 月 日为优先权日，
   专利局的申请日 年 月 日为优先权日，
   专利局的申请日 年 月 日为优先权日。
   ☑申请人已经提交了经原申请国受理机关证明的第一次提出的在先申请文件的副本。
   ☐申请人尚未提交经原申请国受理机关证明的第一次提出的在先申请文件的副本，根据专利法第30条的规定视为未提出优先权要求。
3. ☐经审查，申请人于：
   年 月 日提交的 不符合实施细则第51条的规定；
   年 月 日提交的 不符合专利法第33条的规定；
   年 月 日提交的
4. 审查针对的申请文件：
   ☑原始申请文件。 ☐审查是针对下述申请文件的
   申请日提交的原始申请文件的权利要求第 项、说明书第 页、附图第 页；
   年 月 日提交的权利要求第 项、说明书第 页、附图第 页；
   年 月 日提交的权利要求第 项、说明书第 页、附图第 页；
   年 月 日提交的权利要求第 项、说明书第 页、附图第 页；
   年 月 日提交的说明书摘要， 年 月 日提交的摘要附图。
5. ☐本通知书是在未进行检索的情况下作出的。
   ☑本通知书是在进行了检索的情况下作出的。
   ☑本通知书引用下述对比文献（其编号在今后的审查过程中继续沿用）：

编号	文件号或名称	公开日期（或抵触申请的申请日）
1	EP0376271A2	1990.7.4
2	CN1474624A	1994.8.31

6. 审查的结论性意见：
   ☐关于说明书：

☐申请的内容属于专利法第 5 条规定的不授予专利权的范围。
☐说明书不符合专利法第 26 条第 3 款的规定。
☐说明书不符合专利法第 33 条的规定。
☐说明书的撰写不符合实施细则第 18 条的规定。
☐_____

☑关于权利要求书：
☐权利要求_____不具备专利法第 22 条第 2 款规定的新颖性。
☑权利要求 1-7 不具备专利法第 22 条第 3 款规定的创造性。
☐权利要求_____不具备专利法第 22 条第 4 款规定的实用性。
☑权利要求 8-14 属于专利法第 25 条规定的不授予专利权的范围。
☐权利要求_____不符合专利法第 26 条第 4 款的规定。
☐权利要求_____不符合专利法第 31 条第 1 款的规定。
☐权利要求_____不符合专利法第 33 条的规定。
☐权利要求_____不符合专利法实施细则第 2 条第 1 款关于发明的定义。
☐权利要求_____不符合专利法实施细则第 13 条第 1 款的规定。
☐权利要求_____不符合专利法实施细则第 20 条的规定。
☐权利要求_____不符合专利法实施细则第 21 条的规定。
☐权利要求_____不符合专利法实施细则第 22 条的规定。
☐权利要求_____不符合专利法实施细则第 23 条的规定。
☐_____

上述结论性意见的具体分析见本通知书的正文部分。

7. 基于上述结论性意见，审查员认为：
☐申请人应按照通知书正文部分提出的要求，对申请文件进行修改。
☐申请人应在意见陈述书中论述其专利申请可以被授予专利权的理由，并对通知书正文部分中指出的不符合规定之处进行修改，否则将不能授予专利权。
☑专利申请中没有可以被授予专利权的实质性内容，如果申请人没有陈述理由或者陈述理由不充分，其申请将被驳回。
☐_____

8. 申请人应注意下述事项：
(1) 根据专利法第 37 条的规定，申请人应在收到本通知书之日起的肆个月内陈述意见，如果申请人无正当理由逾期不答复，其申请将被视为撤回。
(2) 申请人对其申请的修改应符合专利法第 33 条的规定，修改文本应一式两份，其格式应符合审查指南的有关规定。
(3) 申请人的意见陈述书和/或修改文本应邮寄或递交国家知识产权局专利局受理处，凡未邮寄或递交给受理处的文件不具备法律效力。
(4) 未经预约，申请人和/或代理人不得前来国家知识产权局专利局与审查员举行会晤。

9. 本通知书正文部分共有 2 页，并附有下述附件：
☑引用的对比文件的复印件共 2 份 12 页。☐_____

审查员：王春艳(3416)
2004 年 8 月 17 日

审查部门 通信审查部

回函请寄：100088 北京市海淀区蓟门桥西土城路 6 号 国家知识产权局专利局受理处收
（注：凡寄给审查员个人的信函不具有法律效力）